

## Pengujian Aplikasi *Doctor to Doctor* Menggunakan Metode *Black Box Testing*

F K Sari Dewi\*<sup>1</sup>, S P Adithama<sup>2</sup>, A T Suhardi<sup>3</sup>

<sup>1,2,3</sup> Program Studi Informatika, Universitas Atma Jaya Yogyakarta

E-mail: findra.dewi@uajy.ac.id\*<sup>1</sup>, stephanie.pamela@uajy.ac.id<sup>2</sup>,  
170709523@students.uajy.ac.id<sup>3</sup>

**Abstrak.** Ketersediaan smartphone mempengaruhi perkembangan aplikasi mobile yang bertujuan membantu manusia dalam aktivitas sehari-hari. Salah satu contohnya adalah aplikasi Doctor to Doctor (D2D) yang membantu dokter memperoleh informasi terbaru seputar kesehatan. Untuk mempertahankan kualitas aplikasi yang nyaman bagi pengguna, diperlukan pengujian yang bertujuan menemukan kesalahan dalam aplikasi. Pengujian dilakukan secara manual dan otomatis dengan menggunakan aplikasi pengujian otomatis Katalon Studio pada 16 dari 42 fungsi aplikasi D2D berdasarkan uji fungsional. Penelitian ini membandingkan tingkat efektivitas pengujian manual dan otomatis, serta kelebihan dan kekurangan dari keduanya. Hasil pengujian menunjukkan persentase keberhasilan pengujian aplikasi D2D sebesar 86,89%, namun 7 dari 16 fungsi belum berjalan dengan baik. Hasil pengujian juga menunjukkan bahwa pengujian otomatis menggunakan Katalon Studio kurang efektif dibandingkan dengan pengujian manual dari segi waktu.

**Kata Kunci:** *Black box testing*; *Doctor to Doctor*; Katalon Studio; pengujian manual; pengujian otomatis

**Abstract.** The presence of smartphones has led to the advancement of mobile applications that aim to assist humans in their daily activities. One example is the Doctor to Doctor (D2D) application, which helps doctors obtain the latest information on health. To maintain the quality of the application for comfortable use by users, testing is necessary to identify errors in the application. The testing was conducted manually and automatically using the Katalon Studio automated testing application for 16 of the 42 functions of the D2D application based on functional tests. This study compares the effectiveness of manual and automatic testing, as well as the advantages and disadvantages of each. The test results showed an 86.89% success rate for the D2D application, but 7 of the 16 functions did not work properly. The results also indicated that automated testing using Katalon Studio was less effective than manual testing in terms of time.

**Keywords:** *Black box testing*; *Doctor to Doctor*; Katalon Studio manual testing, automated testing.

### 1. Pendahuluan

Penjaminan mutu atau *quality assurance* bertujuan untuk menghasilkan standar yang responsif dalam upaya mendapatkan hasil yang berkualitas sehingga dapat menghasilkan hasil positif dan sesuai dengan keinginan

[1][2]. Penjaminan mutu bertujuan untuk mengatasi masalah yang timbul akibat kesalahan program atau bug yang umum dikenal. Masalah ini dapat menyebabkan kerugian bagi perusahaan, mulai dari reputasi yang tercoreng hingga kerugian finansial, tergantung pada tingkat keparahan bug yang terjadi. Sebagai contoh, Tokopedia, sebagai salah satu e-commerce terbesar di Indonesia, pernah mengalami kerugian signifikan ketika data pribadi lebih dari 15 juta pengguna aplikasi mereka bocor. Kejadian ini berdampak negatif tidak hanya secara immaterial, tetapi juga secara material bagi Tokopedia [3][4].

D2D atau Doctor to Doctor adalah sebuah aplikasi yang membantu dokter dalam mencari informasi kesehatan terkini. Aplikasi ini dikembangkan oleh PT. Global Urban Esensial (GUE), anak perusahaan Dexa Group yang fokus pada pengembangan layanan digital di sektor farmasi dan kesehatan. Meskipun aplikasi ini sudah diluncurkan, saat melakukan penambahan fitur atau perubahan pada kode program, diperlukan pengujian regresi. Tujuan dari pengujian regresi adalah memastikan bahwa perubahan atau penambahan tersebut tidak memengaruhi fungsi atau fitur yang sudah ada sebelumnya [5][6]. Metode pengujian regresi dapat dilakukan dengan menggunakan black box testing atau white box testing.

Black box testing merupakan teknik pengujian perangkat lunak di mana pengetahuan detail tentang struktur aplikasi tidak diperlukan. Fokusnya hanya pada aspek-aspek mendasar dari sistem yang berjalan [7]. Pengujian menggunakan black box testing umumnya lebih cepat dalam pelaksanaannya. Pengujian perangkat lunak dengan metode ini memerlukan perangkat lunak dan alat pendukung untuk menjalankan proses otomatisasi. Dalam penelitian ini, Katalon Studio digunakan sebagai salah satu alat untuk menguji otomatis aplikasi D2D, terutama dalam menguji fungsionalitas yang ada. Penelitian ini bertujuan untuk mencapai tingkat keefektifan yang optimal dalam pengujian otomatis, memastikan bahwa fungsionalitas aplikasi D2D sesuai dengan standar yang telah ditetapkan, dan mengevaluasi kelebihan serta kekurangan dari pengujian otomatis dan pengujian manual.

Sejauh ini, telah dilakukan banyak penelitian mengenai pengujian aplikasi oleh beberapa peneliti. Pengujian yang kurang tepat dan kurang diperhatikan seringkali menyebabkan kesalahan dan kerugian saat aplikasi sudah dalam tahap produksi. Oleh karena itu, diperlukan pengujian aplikasi yang tepat untuk mengurangi risiko terjadinya kesalahan atau bug sebelum aplikasi berada di tahap produksi. Beberapa pihak telah melakukan penelitian mengenai pengujian aplikasi *black box testing*. Salah satu contoh penelitian yang dilakukan oleh Muhtadi dan rekan-rekannya adalah “Analisis GUI Testing pada Aplikasi E-Commerce menggunakan Katalon”. Penelitian ini membahas pada aplikasi Tokopedia, JD.ID, dan Bukalapak, di mana peneliti menganalisis hubungan antara kompleksitas GUI (Graphical User Interface) dengan response time. Beberapa fitur yang diuji meliputi onboarding, login, pencarian produk, rincian produk, dan halaman utama produk, dengan menggunakan Katalon Studio sebagai tools pengujian. Dalam lima fitur yang diuji, Bukalapak memiliki response time tercepat secara keseluruhan. Namun, halaman utama Bukalapak memiliki response time terlambat dibandingkan dengan dua aplikasi lainnya. Ini menunjukkan bahwa kompleksitas suatu halaman aplikasi tidak sepenuhnya mempengaruhi response time, karena terdapat beberapa faktor lain yang dapat mempengaruhi response time dari GUI aplikasi [8].

Sebuah penelitian lain tentang *black box testing* telah dilakukan oleh Purnomo yang berjudul “Software Testing Aplikasi Website PT Gramedia menggunakan Metode Black Box pada PT WGS Bandung”. Peneliti melakukan pengujian pada aplikasi berbasis website PT. Gramedia Pustaka Utama dengan metode *black box testing* pada tahap staging perusahaan. Penelitian ini bertujuan untuk mengukur kualitas perangkat lunak dengan berbagai jenis pengujian seperti *functional*, *security*, *front-end performance*, *back-end performance*, dan *user interface* pada aplikasi berbasis web PT. Gramedia Pustaka Utama. Kesimpulan dari penelitian ini terdiri dari lima jenis pengujian. Pengujian *functional* mencakup proses input, output, dan proses aplikasi website telah memenuhi standar. Pengujian *front-end performance* secara keseluruhan memenuhi standar. Pengujian *back-end performance* juga memuaskan dengan data penggunaan aplikasi yang sebagian besar berasal dari luar negeri dan data yang diterima oleh klien lebih

dari 190 MB. Namun, terdapat kelemahan pada jenis URL yang rentan terhadap serangan dari pihak luar, dan dari segi *user interface*, semua kebutuhan telah terpenuhi [9].

Sebuah penelitian dilakukan oleh Cholifah dan timnya dengan judul “Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap”. Aplikasi Action & Strategy merupakan sebuah teknologi berbasis android yang digunakan untuk berbagi pengetahuan bagi 63 lembaga tertentu dan menyimpan sejumlah informasi yang dibutuhkan, dengan kriteria kebenaran, kerahasiaan, dan kelayakan. Aplikasi ini dapat menulis dan menyebarkan informasi ke tujuh platform yang berbeda. Hasil penelitian menunjukkan bahwa data yang tersimpan sesuai dengan masukan yang diberikan, tidak ada kesalahan pada antarmuka, perlu dilakukan penyusunan prosedur pengguna aplikasi, dan keamanan data dapat ditingkatkan dengan penggunaan kata sandi dan pengaturan hak [10].

Krismadi dan rekannya melakukan pengujian kualitas aplikasi website dengan menggunakan metode *black box* berbasis *equivalent partitioning*. Penelitian ini berjudul “Pengujian Black Box berbasis Equivalence Partitioning Pada Aplikasi Promosi Kenaikan Jabatan”. Penelitian ini melibatkan tiga model pengelompokan kesalahan, yaitu fungsi, struktur, dan interface. Hasilnya menunjukkan adanya kesalahan dalam menghapus data yang tersimpan. Secara keseluruhan, pengujian aplikasi Seleksi Promosi Kenaikan Jabatan dapat membantu proses pembuatan *case* pengujian, meningkatkan kualitas aplikasi, dan menemukan kesalahan yang tidak disengaja [11].

Wahyuji dan timnya melakukan penelitian dengan judul “Pengembangan dan Pengujian Aplikasi Website Career Center ITERA”. Penelitian ini menggunakan Katalon Studio sebagai alat pengujian otomatis. Tujuan dari penelitian ini adalah untuk menghasilkan sistem penyedia lowongan pekerjaan yang terintegrasi berbasis web. Dalam penelitian ini, didapatkan total waktu respon sebesar 159,4 detik atau rata-rata 14,5 detik. Terdapat kesalahan dalam pengisian tanggal pada aplikasi karena Katalon Studio tidak dapat mendeteksi masukan berupa datepicker pada aplikasi web. Meskipun ada kesalahan, fungsi Katalon Studio sebagai alat pengujian otomatis telah terpenuhi [12].

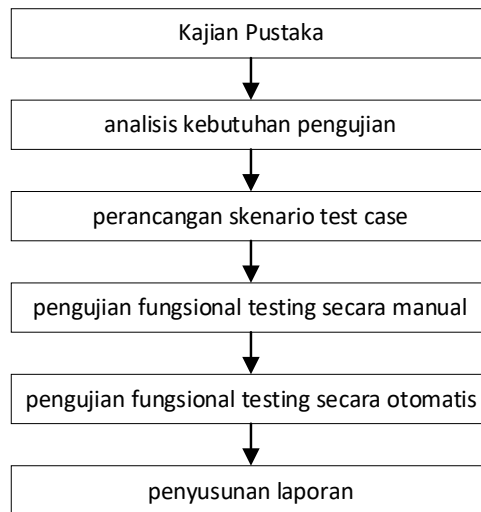
Penelitian lain yang dilakukan oleh Fitria dan kawan-kawan adalah “Pengujian Black Box pada Aplikasi E-Commerce berbasis Website Menggunakan Metode Boundary Value Analysis dan Equivalence Partitioning”. Penelitian ini bertujuan untuk menguji kualitas aplikasi E-Commerce pada website dengan metode *boundary value analysis* dan *equivalence partitioning*. Hasil dari penelitian ini menunjukkan bahwa pengujian dengan metode *boundary value analysis* dan *equivalence partitioning* dapat membantu menemukan kesalahan pada aplikasi *e-commerce* dan meningkatkan kualitas aplikasi secara keseluruhan [13].

Penelitian terakhir yang dapat disebutkan adalah penelitian yang dilakukan oleh Nurhakim dan kawan-kawan dengan judul “Pengujian Black Box pada Aplikasi Pengenalan Jenis Buah-buahan menggunakan Metode Equivalence Partitioning”. Penelitian ini bertujuan untuk menguji kualitas aplikasi pengenalan jenis buah-buahan dengan metode *equivalence partitioning*. Hasil dari penelitian ini menunjukkan bahwa pengujian dengan metode *equivalence partitioning* dapat membantu menemukan kesalahan pada aplikasi pengenalan jenis buah-buahan dan meningkatkan kualitas aplikasi secara keseluruhan [14].

## **2. Metode**

Penelitian ini menggunakan beberapa metode penelitian, yaitu kajian pustaka, analisis kebutuhan pengujian, perancangan skenario *test case*, pengujian fungsional testing secara manual, pengujian fungsional secara otomatis, dan penyusunan laporan. Kajian pustaka dilakukan selama satu bulan dengan mencari informasi dari buku, karya ilmiah, tugas akhir, internet, dan sumber lain yang berkaitan dengan penelitian. Tahap analisis kebutuhan pengujian digunakan untuk mempelajari *use case* sehingga dapat dirancang *test case*. Selanjutnya, perancangan skenario *test case* dilakukan untuk menulis *test case*

berdasarkan analisis sebelumnya. Pada tahap pengujian fungsional testing secara manual, aplikasi diuji berdasarkan fungsi dan alur yang telah ditulis dalam skenario, tanpa menggunakan *automation testing tools*. Hasil pengujian dicatat dalam laporan. Pada tahap pengujian fungsional testing secara otomatis, aplikasi diuji berdasarkan fungsi dan alur yang telah ditulis dalam skenario, menggunakan *automation testing tools* Katalon Studio. Hasil pengujian juga dicatat dalam laporan. Alur metodologi penelitian dapat dilihat pada Gambar 1.



**Gambar 1.** Diagram Alur Metodologi Penelitian

### 3. Hasil dan Diskusi

#### 3.1. Analisis Kebutuhan Pengujian

Sebelum melakukan pengujian secara manual, langkah yang perlu dilakukan adalah melakukan analisis alur fungsi dari aplikasi Doctor to Doctor. Pada tahap ini, peneliti akan memposisikan diri sebagai pengguna aplikasi yang sedang menggunakannya dengan mempertimbangkan dokumen kebutuhan produk (Product Requirement Document/PRD) serta *test case* yang tersedia. Tujuannya adalah agar penulis dapat mengenali setiap fungsi aplikasi dan membandingkannya dengan spesifikasi yang diinginkan. Pengujian secara otomatis memiliki struktur yang hampir sama dengan pengujian manual, namun dibantu dengan penggunaan *automation testing tools* berdasarkan *test case* yang telah dibuat sebelumnya. Pada penelitian ini, *automation testing tool* yang digunakan adalah Katalon Studio versi 7.9.1. Selain itu, perangkat keras yang digunakan selama pengujian meliputi *smartphone*, laptop/komputer, dan kabel data untuk mentransfer data atau sebagai penghubung antara laptop dan perangkat mobile. Pemilihan Katalon Studio didasarkan pada kemudahan penggunaan dan antarmuka pengguna yang bersahabat dari *automation testing tool* tersebut.

#### 3.2. Perancangan Pengujian

Pada penelitian ini, *use case* yang dibuat untuk diuji cukup banyak, yaitu *use case* pilih negara, *use case* login aplikasi, *use case* registrasi aplikasi, *use case* mencari webinar, *use case* mencari informasi obat, *use case* mencari event, *use case* request jurnal, *use case* mencari konten di *homepage*, *use case* mengubah detail profil dokter, *use case* mengubah *header* profil dokter, *use case* mencari jurnal, *use case* mencari *guideline*, *use case* mencari video kesehatan, *use case* mencari channel, *use case* memasukan komentar channel forum dan *use case* channel request jurnal. Pada manuskrip ini hanya akan ditampilkan salah satu *use case* saja, yaitu *use case* mencari jurnal.

### 3.2.1. Perancangan Test Case

Contoh yang akan diberikan adalah perancangan *test case* untuk *use case* mencari jurnal.

Deskripsi: Aktor mencari dan membuka detail jurnal sesuai dengan pencarian.

Aktor : Pengguna Aplikasi Doctor to Doctor

*Preconditions:*

1. Aktor sudah login aplikasi
2. Aktor berada di halaman utama aplikasi

*Postconditions:*

1. Pencarian jurnal ditemukan
2. Detail jurnal yang dipilih berhasil ditampilkan

#### a. Basic Flow

1. *Use case* dimulai ketika aktor berada di halaman utama aplikasi
2. Aktor mengetuk ikon literatur
3. Sistem menampilkan halaman spesialisasi
4. Aktor memilih salah satu spesialisasi yang diinginkan
5. Sistem menampilkan halaman literasi jurnal serta list dokumen pustaka
6. Aktor mengetuk ikon pencarian di bagian atas aplikasi
7. Aktor memasukkan inputan pencarian jurnal

A1 : Terdapat kolom yang tidak sesuai dengan ketentuan

1. Sistem menampilkan list jurnal sesuai dengan inputan aktor berdasarkan judul
2. Aktor mengetuk jurnal yang dipilih
3. Sistem menampilkan detail jurnal
4. Aktor ketuk kembali ke halaman list dokumen pustaka
5. *Use case* selesai

#### b. Alternative sequence

A1 : Terdapat kolom yang tidak sesuai dengan ketentuan.

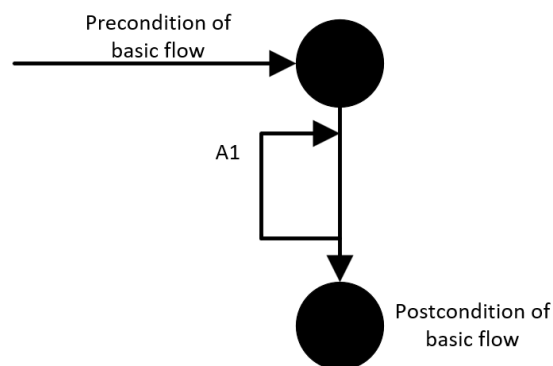
A1 dimulai pada Langkah 7 *basic flow*

8. Pada kolom inputan yang tidak sesuai, sistem akan menampilkan informasi kegagalan. Sistem akan menampilkan informasi kegagalan berupa warning dalam aplikasi ‘Pencarian harus spesifik min. 3 karakter’

Kembali ke *basic flow* langkah 7.

#### c. Matriks skenario untuk *flow of event*

Bagan alur dan tabel skenario untuk *use case* mencari jurnal dapat dilihat pada Gambar 2 dan Tabel 1.



**Gambar 2.** Bagan Alur *Use Case* Mencari Jurnal

**Tabel 1.** Tabel Skenario *Use Case* Mencari Jurnal

S ID	Skenario Pengujian	Keterangan
S1	Skenario 1: BF	Berjalan Normal
S2	Skenario 2: BF A1	Terdapat kolom yang tidak sesuai dengan ketentuan

**d. Identifikasi Variabel**

Nama variabel: Pencarian.

Deskripsi: Merupakan inputan pencarian oleh aktor untuk melakukan pencarian jurnal berdasarkan nama atau title.

**e. Validity Check**

VC ID: VC1

Deskripsi: Input pencarian tidak boleh kurang dari 3 karakter.

**f. Matriks Test Case**

Matriks *test case* untuk *use case* mencari jurnal dapat dilihat pada Tabel 2.

**Tabel 2.** Matriks *Test Case* untuk *Use Case* Mencari Jurnal

TC ID	S ID	Pencarian	Expected Output	Keterangan
TC1	S1	valid	Berhasil	Pencarian berhasil dilakukan.
TC2	S2	invalid	Gagal	Pencarian < 3 karakter

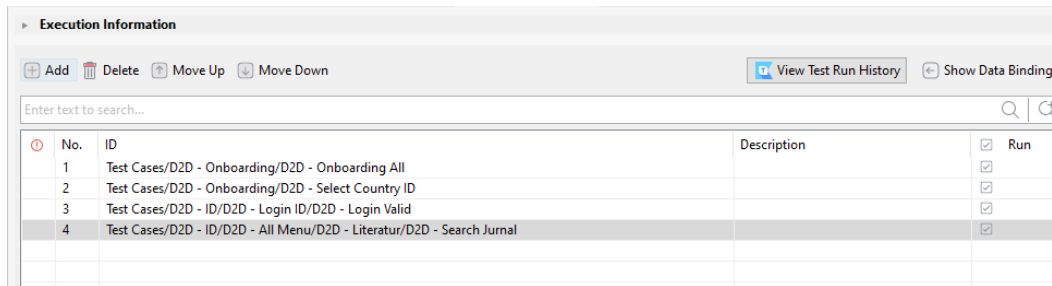
**3.3. Analisis Hasil Pengujian Otomatis**

Pada bagian ini juga hanya akan ditampilkan contoh hasil pengujian otomatis untuk *use case* mencari jurnal saja. Gambar 3 menunjukkan cara menggunakan aplikasi untuk mencari jurnal. Pertama-tama, pengguna harus menyetuk ikon literature pada *homepage*, dan memilih spesialisasi dokter. Pada pengujian ini, spesialisasi yang digunakan adalah Dokter Umum. Selanjutnya, pengguna harus menyetuk ikon pencarian dan memilih opsi pencarian jurnal. Setelah itu, pengguna dapat memberikan inputan ke dalam kolom pencarian. Ada dua jenis inputan, yaitu valid dan invalid. Inputan valid dideklarasikan secara *hardcode* pada Katalon Studio, sementara inputan *invalid* digunakan untuk memverifikasi *output warning* yang muncul ketika pengguna menginput kurang dari tiga karakter.

Item	Object	Input
1 - Tap	Submenu - Literatur	0
2 - Tap	android.widget.TextView - Dokter Un	0
3 - Tap	android.widget.TextView -	0
4 - Tap	android.widget.TextView - JOURNAL	0
5 - Set Text	android.widget.EditText - Search vide	"Testing"; 0
6 - Scroll To Text		"Testing Deskripsi Jurnal Indo"
7 - Tap	android.widget.TextView - Testing De	0
8 - Press Back		
9 - Tap	android.widget.TextView -	0
10 - Set Text	android.widget.EditText - Search vide	"Qwerty"; 0
11 - Tap	android.widget.TextView -	0
12 - Set Text	android.widget.EditText - Search vide	"Te"; 0
13 - Tap	android.widget.TextView -	0
14 - Scroll To Text		"Pencarian harus spesifik min. 3 kara
15 - Press Back		

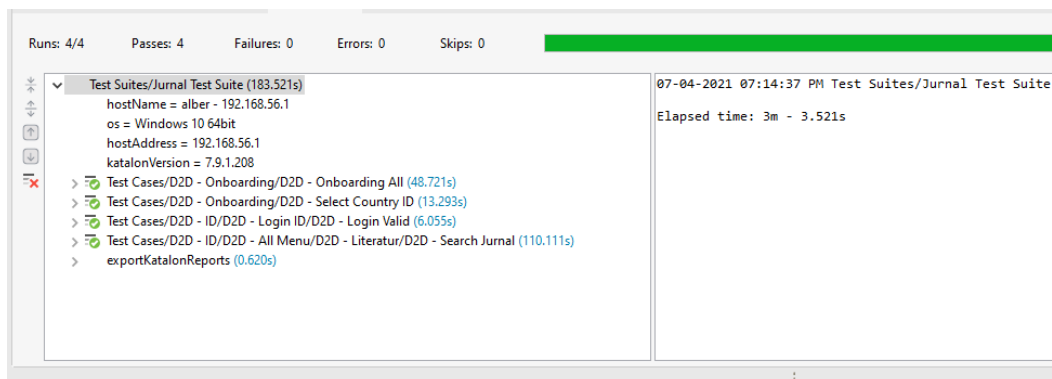
**Gambar 3.** Test Case Katalon Studio untuk *Use Case* Mencari Jurnal

Setelah rancangan selesai dibuat, dilakukan pengujian otomatis menggunakan *test suite* pada Gambar 4. Hasil pengujian menunjukkan bahwa terdapat satu langkah yang gagal dijalankan pada langkah ke-14 (Gambar 7). Namun, karena menggunakan Failure Handling Optional, pengujian masih dapat dilanjutkan dengan notifikasi *warning*. Hal ini menunjukkan bahwa ada kesalahan pada sistem yang tidak memberikan peringatan saat inputan kurang dari tiga karakter.



**Gambar 4.** Test Suite Katalon Studio untuk *Use Case* Mencari Jurnal

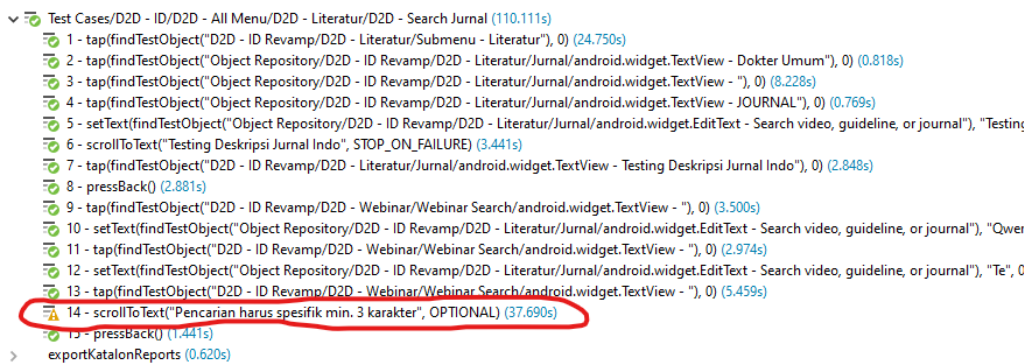
Hasil pengujian keseluruhan menunjukkan bahwa fungsionalitas mencari jurnal otomatis masih belum berjalan dengan baik, dengan presentase pengujian sebesar 66,7%. Hasil pengujian dicatat pada log viewer Katalon dan website test ops Katalon, yang ditunjukkan pada Gambar 5 dan Gambar 6.



**Gambar 5.** Hasil Pengujian Otomatis untuk *Use Case* Mencari Jurnal

Status	Name	Maintainer	Assertions	Last Test Run	Average Duration	Fliakiness (%)	Requirements
✓	D2D - Search Jurnal Test Cases/D2D - ID/D2D - All Menu/D2D - Literatur		1	⌚ 4 mins ago ⌚ 1m 50s	1m 50s	0.00	
✓	D2D - Login Valid Test Cases/D2D - ID/D2D - Login ID		0	⌚ 4 mins ago ⌚ 6s	9s	13.33	
✓	D2D - Select Country ID Test Cases/D2D - Onboarding		0	⌚ 5 mins ago ⌚ 13s	11s	12.50	
✓	D2D - Onboarding All Test Cases/D2D - Onboarding		0	⌚ 5 mins ago ⌚ 49s	44s	12.50	

**Gambar 6.** Hasil Pengujian Otomatis pada Katalon TestOps untuk *Use Case* Mencari Jurnal



Gambar 7. Hasil Pengujian Otomatis untuk Use Case Mencari Jurnal

### 3.4. Persentase Keberhasilan Pengujian

Hasil pengujian aplikasi Doctor to Doctor (D2D) menunjukkan bahwa dari 16 fungsi yang diuji, rata-rata persentase keberhasilannya sebesar 86,89%. Untuk menghitung persentase keberhasilan masing-masing *use case*, digunakan rumus banyaknya *test case* yang berhasil dibagi dengan jumlah total *test case pada use case* tersebut, kemudian hasilnya dikalikan dengan 100%. Hasil persentase keberhasilan masing-masing *use case* ditampilkan pada Tabel 1. Sedangkan persentase keberhasilan total dihitung dengan menjumlahkan semua persentase keberhasilan *use case* dan dibagi dengan jumlah total *use case*, hasilnya juga tertera di Tabel 1. Berdasarkan skala penilaian menurut Arikunto yang tercantum pada Tabel 2, aplikasi D2D dapat dikategorikan sangat baik.

Tabel 1. Persentase Keberhasilan Pengujian Aplikasi D2D

No	Fungsi	Persentase Keberhasilan	Test Case Berhasil	Test Case Gagal	Total Test Case
1.	Pilih Negara	100%	2	0	2
2.	Login Aplikasi	100%	6	0	6
3.	Registrasi Aplikasi	100%	9	0	9
4.	Mencari Webinar	66,7%	2	1	3
5.	Mencari Informasi Obat	66,7%	2	1	3
6.	Mencari <i>Event</i>	66,7%	2	1	3
7.	Request Jurnal	100%	7	0	7
8.	Mencari Konten di Homepage	100%	3	0	3
9.	Mengubah Detail Profil Dokter	90%	9	1	10
10.	Mengubah Header Profil Dokter	100%	6	0	6
11.	Mencari Jurnal	66,7%	2	1	3
12.	Mencari <i>Guideline</i>	66,7%	2	1	3
13.	Mencari Video Kesehatan	66,7%	2	1	3
14.	Mencari Channel	100%	3	0	3
15.	Menginput Komentar Channel Forum	100%	3	0	3
16.	Channel Request Jurnal	100%	3	0	3
Total		1390,2%	67	7	74
Rata-rata			86,89%		



**Tabel 2.** Kategori Kelayakan Kualitas Sistem Informasi Menurut Arikunto[15]

Kategori	Persentase
Sangat Baik	81% - 100%
Baik	61% - 80%
Cukup Baik	41% - 60%
Tidak Baik	21% - 40%
Sangat Tidak Baik	< 20%

### 3.5. Perbandingan Hasil Pengujian Manual dan Otomatis

Tabel 3 menunjukkan hasil pengujian manual dan otomatis berdasarkan waktu pengujian. Ada 13 fungsi yang lebih cepat diuji secara manual daripada otomatis. Namun, ada tiga fungsi seperti pemilihan negara, login, dan registrasi yang lebih cepat diuji secara otomatis. Total selisih waktu dari 13 fungsi yang diuji secara manual lebih cepat dari otomatis adalah 11 menit 18 detik, sedangkan total selisih waktu dari 3 fungsi yang diuji secara otomatis lebih cepat dari manual adalah 1 menit 51 detik.

**Tabel 3.** Perbandingan Waktu Pengujian Manual, Waktu Otomatis Sebenarnya, dan Selisih Waktu

No	Fungsi	Waktu Pengujian		Selisih Waktu
		Manual	Otomatis	
1	Pilih Negara	26 detik	24 detik	-2 detik
2	Login Aplikasi	38 detik	23 detik	-15 detik
3	Registrasi Aplikasi	2 menit 25 detik	51 detik	-1 menit 34 detik
4	Mencari Webinar	25 detik	1 menit 22 detik	1 menit 20 detik
5	Mencari Informasi Obat	20 detik	1 menit 36 detik	1 menit 16 detik
6	Mencari <i>Event</i>	28 detik	1 menit 44 detik	1 menit 16 detik
7	Request Jurnal	1 menit 15 detik	1 menit 55 detik	40 detik
8	Mencari Konten di Homepage	20 detik	55 detik	35 detik
9	Mengubah Detail Profil Dokter	1 menit 42 detik	2 menit 8 detik	26 detik
10	Mengubah Header Profil Dokter	49 detik	56 detik	7 detik
11	Mencari Jurnal	27 detik	1 menit 50 detik	1 menit 23 detik
12	Mencari <i>Guideline</i>	25 detik	1 menit 43 detik	1 menit 18 detik
13	Mencari Video Kesehatan	23 detik	1 menit 40 detik	1 menit 17 detik
14	Mencari Channel	26 detik	49 detik	23 detik
15	Menginput Komentar Channel Forum	35 detik	1 menit 21 detik	46 detik
16	Channel Request Jurnal	1 menit 20 detik	1 menit 51 detik	31 detik

Persamaan (1) adalah rumus perhitungan penurunan waktu eksekusi pengujian otomatis dibandingkan dengan pengujian manual.

$$\text{Persentase Penurunan Waktu} = \frac{\text{Waktu Pengujian Otomatis} - \text{Waktu pengujian Manual}}{\text{Waktu Pengujian Manual}} \times 100\% \quad (1)$$

Tabel 4 menunjukkan hasil pengujian fungsi dengan perhitungan rata-rata persentase penurunan waktu eksekusi pada Gambar 1. Dari 16 fungsi yang diuji, rata-rata total persentase penurunan waktu eksekusi pada pengujian otomatis adalah 140,524%. Terdapat tiga fungsi yang pengujian otomatis lebih cepat dengan rata-rata persentase penurunan waktu eksekusi sebesar -37,331%. Namun, pada 13 fungsi lainnya, pengujian otomatis lebih lambat daripada pengujian manual dengan rata-rata persentase penurunan waktu eksekusi sebesar 181,567%. Rata-rata pengujian otomatis lebih cepat 0,627x dari pengujian manual pada tiga fungsi, sementara pada 13 fungsi lainnya, rata-rata pengujian otomatis lebih lambat 2,816x dari pengujian manual.

**Tabel 4.** Persentase Penurunan Waktu Eksekusi Pengujian Otomatis dibandingkan Pengujian Manual

No	Fungsi	Waktu Pengujian		Penurunan Waktu Eksekusi (%)	Keterangan
		Manual	Otomatis		
1	Pilih Negara	26	24	-7.692%	Pengujian Otomatis 0,923 kali Lebih Cepat
2	Login Aplikasi	38	23	-39.473%	Pengujian Otomatis 0,605 kali Lebih Cepat
3	Registrasi Aplikasi	145	51	-64.827%	Pengujian Otomatis 0,352 kali Lebih Cepat
4	Mencari Webinar	25	82	228%	Pengujian Otomatis 3,28 kali Lebih Lambat
5	Mencari Informasi Obat	20	96	380%	Pengujian Otomatis 4,8 kali Lebih Lambat
6	Mencari <i>Event</i>	28	104	271,428%	Pengujian Otomatis 3,714 kali Lebih Lambat
7	Request Jurnal	75	115	53,333%	Pengujian Otomatis 1,533 kali Lebih Lambat
8	Mencari Konten di Homepage	20	55	175%	Pengujian Otomatis 2,75 kali Lebih Lambat
9	Mengubah Detail Profil Dokter	102	128	25,490%	Pengujian Otomatis 1,255 kali Lebih Lambat
10	Mengubah Header Profil Dokter	49	56	14,286%	Pengujian Otomatis 1,143 kali Lebih Lambat
11	Mencari Jurnal	27	110	307,407%	Pengujian Otomatis 4,074 kali Lebih Lambat
12	Mencari <i>Guideline</i>	25	103	312%	Pengujian Otomatis 4,12 kali Lebih Lambat
13	Mencari Video Kesehatan	23	100	334,783%	Pengujian Otomatis 4,348 kali Lebih Lambat
14	Mencari Channel	26	49	88,462%	Pengujian Otomatis 1,885 kali Lebih Lambat
15	Menginput Komentar Channel Forum	35	81	131,429%	Pengujian Otomatis 2,314 kali Lebih Lambat
16	Channel Request Jurnal	80	111	38,75%	Pengujian Otomatis 1,387 kali Lebih Lambat
Rata-rata				140,524%	Rata-rata pengujian otomatis 13 fungsi yang lain rata-rata pengujian otomatis 2,816 kali lebih lambat

Persamaan (2) adalah rumus perhitungan peningkatan kecepatan pengujian otomatis dibandingkan dengan pengujian manual.

$$\text{Persentase Peningkatan Kecepatan} = \frac{\left(\frac{1}{\text{waktu pengujian otomatis}} - \frac{1}{\text{waktu pengujian manual}}\right)}{\frac{1}{\text{waktu penilaian manual}}} \times 100\% \quad (2)$$

Berdasarkan hasil di Tabel 5 dan perhitungan rata-rata persentase penurunan waktu eksekusi pada Gambar 2, dari 16 fungsi, terdapat tiga fungsi yang pengujian otomatisnya lebih cepat dengan rata-rata persentase penurunan waktu eksekusi sebesar -37,331%. Namun, pada 13 fungsi lainnya, pengujian otomatis justru lebih lambat dibandingkan dengan pengujian manual, dengan rata-rata persentase penurunan waktu eksekusi sebesar 181,567%. Dalam tiga fungsi yang lebih cepat tersebut, pengujian otomatis rata-rata 0,627x lebih cepat daripada pengujian manual, sementara pada 13 fungsi lainnya, pengujian otomatis rata-rata 2,816x lebih lambat.

**Tabel 5.** Persentase Peningkatan Kecepatan Pengujian Otomatis dibandingkan Pengujian Manual

No	Fungsi	Waktu Pengujian		Peningkatan Kecepatan (%)
		Manual	Otomatis	
1	Pilih Negara	26	24	8,333
2	Login Aplikasi	38	23	65,217
3	Registrasi Aplikasi	145	51	184,314
4	Mencari Webinar	25	82	-69,512
5	Mencari Informasi Obat	20	96	-79,167
6	Mencari <i>Event</i>	28	104	-73,077
7	Request Jurnal	75	115	-34,783
8	Mencari Konten di Homepage	20	55	-63,636
9	Mengubah Detail Profil Dokter	102	128	-20,313
10	Mengubah Header Profil Dokter	49	56	-12,5
11	Mencari Jurnal	27	110	-75,455
12	Mencari <i>Guideline</i>	25	103	-75,728
13	Mencari Video Kesehatan	23	100	-77
14	Mencari Channel	26	49	-46,938
15	Menginput Komentar Channel Forum	35	81	-56,790
16	Channel Request Jurnal	80	111	-27,928
Rata-rata				-28,435

Dari hasil pengujian dan data Tabel 5 di atas dapat disimpulkan bahwa pengujian manual unggul dari segi waktu dibandingkan pengujian otomatis. Salah satu faktor yang mempengaruhi pengujian manual unggul pada 13 fungsi adalah adanya waktu tunda bagi Katalon Studio dalam mengenali objek yang akan dikenakan instruksi. Sedangkan pengujian otomatis unggul pada 3 fungsi yaitu pilih negara, login aplikasi, dan registrasi aplikasi.

#### 4. Kesimpulan

Dari hasil penelitian dan pembahasan yang dilakukan, ditemukan bahwa dari 16 fungsi yang diuji, rata-rata persentase keberhasilan pengujian adalah 86,89%. Berdasarkan kriteria kelayakan menurut Arikunto, aplikasi D2D telah dikategorikan sebagai sangat baik. Meskipun begitu, masih ditemukan 7 *bug* pada 7

fungsi yang diuji yang memerlukan perbaikan oleh pengembang agar kualitas aplikasi D2D menjadi lebih baik. Dalam hal pengujian, pengujian otomatis menggunakan Katalon Studio terbukti kurang efektif dalam hal waktu dibandingkan dengan pengujian manual. Dari 16 fungsi yang diuji, 3 fungsi memiliki selisih waktu pengujian otomatis yang lebih cepat selama 1 menit 51 detik dibandingkan dengan pengujian manual. Namun, untuk 13 fungsi lainnya, selisih waktu pengujian manual lebih cepat 10 menit 55 detik dari pengujian otomatis. Hal ini disebabkan oleh waktu tunda Katalon Studio dalam mendeteksi objek yang akan diberi instruksi.

## Referensi

- [1] A. D. Galang, I. V. Ancho, U. Hindu, N. I. Gusti, and B. Sugriwa. (2020). “Jurnal Penjaminan Mutu Corpus Thematic Analysis: Trends In Quality And Quality Assurance Research,”. *Jurnal Penjaminan Mutu Lembaga Penjaminan Mutu Institusi Hindu Dharma Negeri Denpasar*, vol. 6, pp. 124–133.
- [2] Sutikno, B. (2016). Quality Assurance of Information Systems Development in the Context of Higher Education. *Journal of Theoretical and Applied Information Technology*, 93(2), 360-371.
- [3] Pertiwi, Wahyunanda Kusuma.(2020).*Data 15 Juta Pengguna Diduga Bocor, Tokopedia Sebut Ada Upaya Pencurian.*” <https://tekno.kompas.com/read/2020/05/02/22060847/data-15-juta-pengguna-diduga-bocor-tokopedia-sebut-ada-upaya-pencurian> (accessed Jan. 31, 2021).
- [4] Tempo.co. (2020). Data Pengguna Tokopedia Dibocorkan, Ini Sejumlah Klarifikasi, (Online), <https://bisnis.tempo.co/read/1412511/data-pengguna-tokopedia-dibocorkan-ini-sejumlah-klarifikasi>, diakses 24 Agustus 2022.
- [5] P. Kandil, S. Moussa, and N. Badr.(2015).“A Study for Regression Testing Techniques and Tools,” *Int. J. Soft Comput. Softw. Eng.*, vol. 5, no. 4, pp. 64-84.
- [6] Kaner, C., Falk, J., & Nguyen, H. Q. (2014). *Testing computer software*. John Wiley & Sons.
- [7] M. Ehmer and F. Khan. (2012) “A Comparative Study of White Box, Black Box and Grey Box Testing Techniques,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 3, no. 6, pp. 12-15.
- [8] Pressman, R. S. (2005). *Software engineering: a practitioner's approach*. Palgrave Macmillan.
- [9] V. Febrian, M. R. Ramadhan, M. Faisal, and A. Saifudin, “Pengujian pada Aplikasi Penggajian Pegawai dengan menggunakan Metode Blackbox,” *J. Inform. Univ. Pamulang*, 2020, vol. 5, no. 1, pp. 61-66.
- [10] M. M. Muhtadi, M. D. Friyadi, and A. Rahmani. (2019). “Analisis GUI Testing pada Aplikasi E-Commerce menggunakan Katalon,” *Pros. Ind. Res. Work. Natl. Semin.*, vol.10 ,no. 1, pp.1387-1393.
- [11] W. N. Cholifah, Y. Yulianingsih, and S. M. Sagita. (2018) “Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap,” *STRING (Satuan Tulisan Ris. dan Inov. Teknol.*, vol. 3,no. 2, pp. 206-210.
- [12] A. Krismadi, A. F. Lestari, A. Pitriyah, I. W. P. A. Mardangga, M. Astuti, and A. Saifudin.(2019).“Pengujian Black Box berbasis Equivalence Partitions pada Aplikasi Seleksi Promosi Kenaikan Jabatan,” *J. Teknol. Sist. Inf. dan Apl.*,vol. 2, no. 4, pp. 155-161.
- [13] Wahyuj, A. T. (2021). Pengembangan dan Pengujian Aplikasi Website Career Center ITERA. *Jurnal Tekno Kompak*, 15(1), 67-78.
- [14] F. Fakhruddin. (2023). “Pengujian Black Box Pada Aplikasi Berbasis Web Menggunakan Metode Boundary Value Analysis dan Equivalence Partitioning (Studi Kasus: Aplikasi E-commerce Easea.co) (Doctoral dissertation, UPN" Veteran" Jawa Timur).
- [15] A. Suhari Camara M, K. Aelani, and F. Dwi Juniar S.(2021).“Pengujian Kualitas Website menggunakan Metode McCall Software Quality,” *J. Inf. Technol.*, vol. 3, no. 1, pp. 25–32.